CLCS Application Software
IPT Kick-off

8/20/97

United Space Alliance

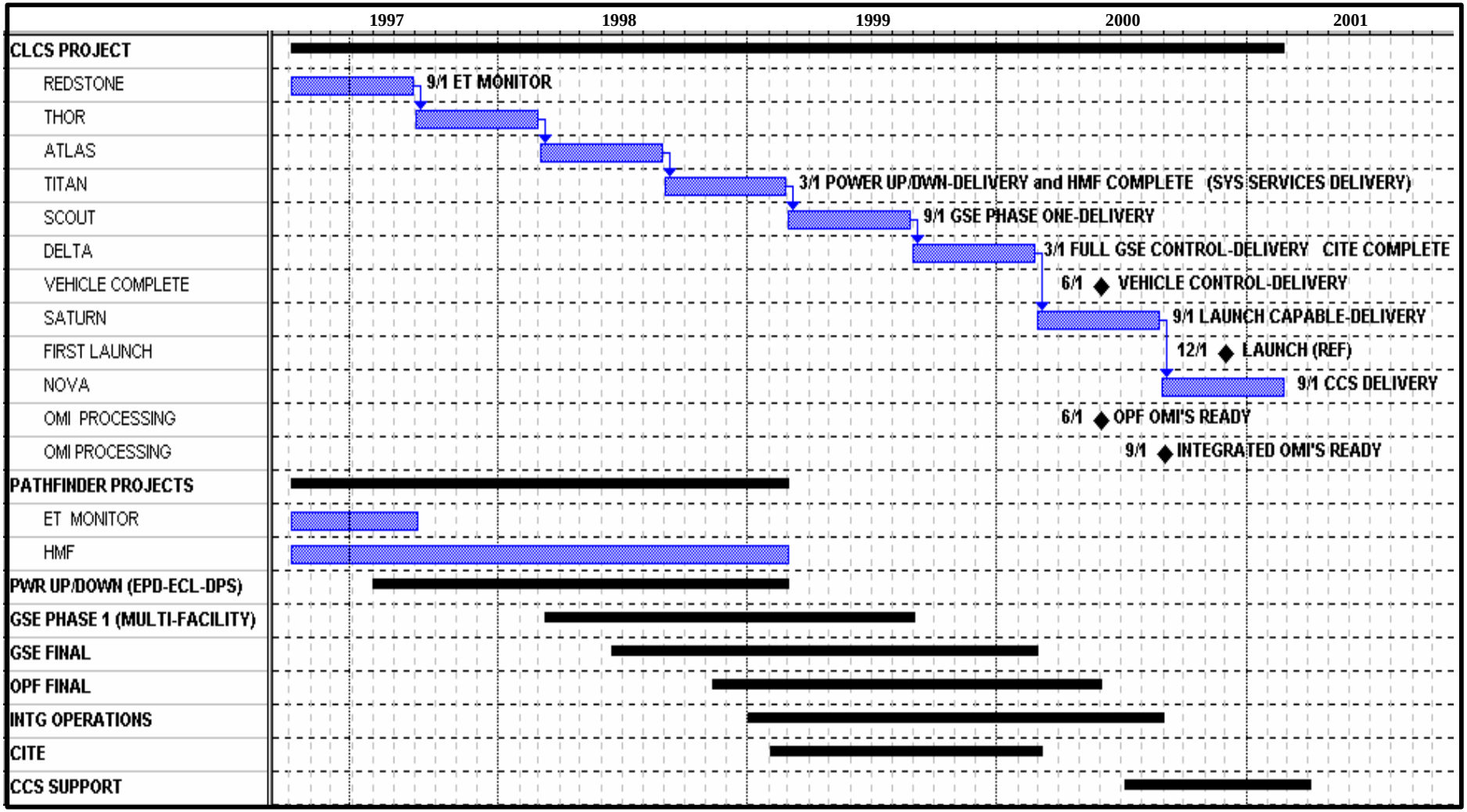
Overview

- Meeting overview
  - Introductions
  - Set the stage
    - perspectives
    - functions
  - Identify the processes
  - Define the mission (a charter)
  - Assign tasks
- Introductions
  - The APTeam
    - Robert Pierce
    - Rich Ikerd
    - Ken Hale
  - Software Achitecture Team (SAT) interface (Debbie Lee)
  - The Integrated Product Team (IPT)
- Questions and comments are ***encouraged***

A CLCS Perspective

- What is this thing called CLCS?
  - CLCS (Checkout and Launch Control System) is the CCMS replacement
  - Ten deliveries (rocket code names) every 6 months (March, Sept)
  - Concept of build a little test a little
  - Redefines operations concept
    - Vastly improved screen capabilities
    - RSYS “function” changes
    - Console position supports
      - 1 command and control workstation (2-20” screens)
      - Business and Information Network (BIN) workstation
- Where does Application Software (APP S/W) fit?
  - Project level decision NOT to translate existing code
  - Functional design concept (Object Oriented)
  - Displays disassociated from control process
  - End Item Mananger (EIM) function
  - Extensive use of COTS (Commercial Off The Shelf) development tools
  - New process for Application development / sustaining

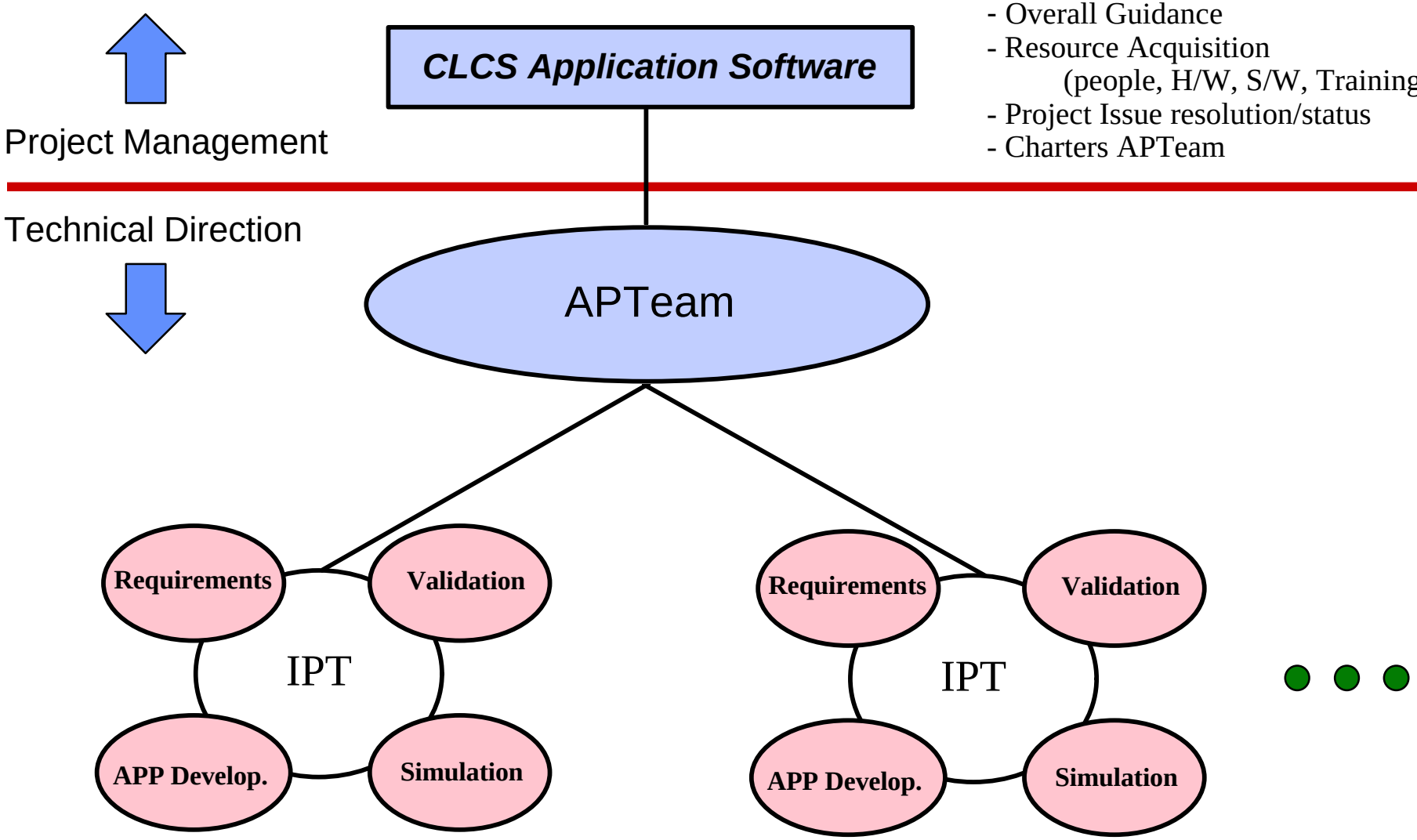
CLCS Deliveries



Note: A more detailed APP S/W delivery schedule is included as reference at the end of this presentation

United Space Alliance

APP Support Structure



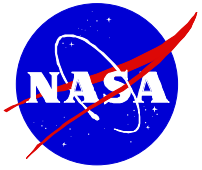
United Space Alliance

The APTeam

- What it does
  - Charters the IPT and establishes core participants
    - Uses master APP S/W schedule as input
    - Accounts for system services deliveries
    - Identifies a “focal point” for each major process
  - Works big picture items
    - Allocation of resources
    - Cross IPT issues
    - CLCS issues
  - Provides common processes, tools, directions (ref: CLCS Software Development Plan)
- Who it reports to
  - APP S/W project management (Ben Bryant, Jim Hurst)
    - Overall guidance and resource acquisition
  - User Liaison (Jeff Wheeler, Chris Best)
    - Non Application User issues (Operations, Plots, Console Layout, User Access, etc.)
    - Resource acquisition
  - These report to CLCS project management (Retha Hart, Ralph Esposito)

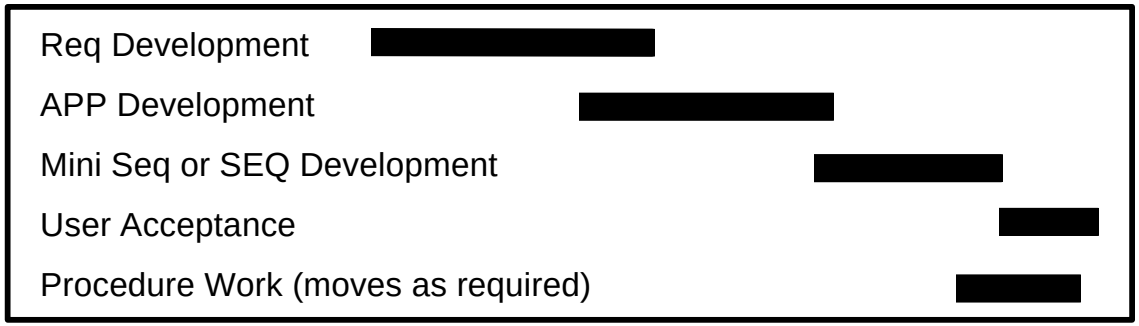
The IPT

- What is an APP S/W IPT?
  - Team responsible for the development and delivery of specific capabilities
  - Multitalented, close knit group
  - Breaks the “Us” versus “Them” barrier
- IPT concepts
  - Core Members
    - Systems Knowledge & Software Production (NASA-USA, others as required)
    - Core members participate in *all* activities of the IPT
    - A focal point for each IPT function will be established (to share the load)
    - Development groups “spun off” to perform a specific task
  - Responsibility
    - Overall lead is responsible to assure the “‘big” plan remains on-track
    - *Everyone* is responsible for the outcome (team effort is critical)
    - Communication is essential
      - Problem areas
      - Successes
  - Efficiency/Teaming enhanced by providing a “common working area”

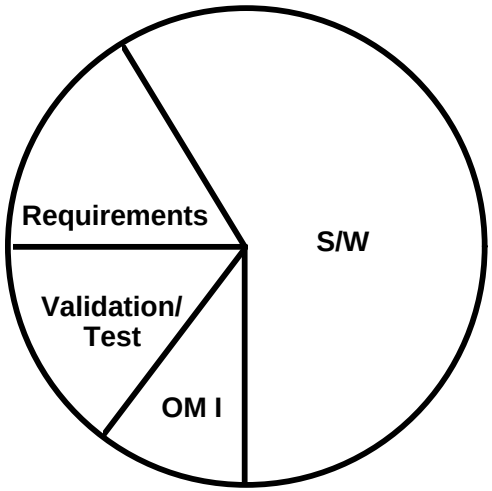


APP S/W Development

- CLCS APP S/W Development (typical)



- Total Level of Effort

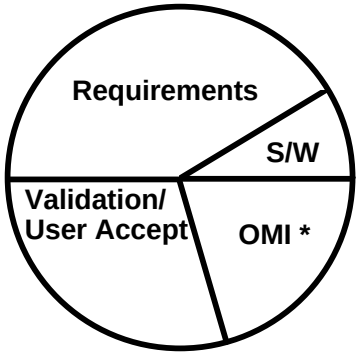


Note: Pie charts reflect effort not number of people

United Space Alliance

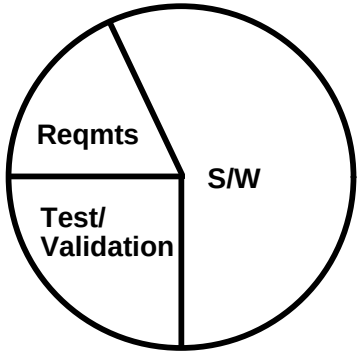
APP S/W Support

- APP S/W Development Support (level of effort %)



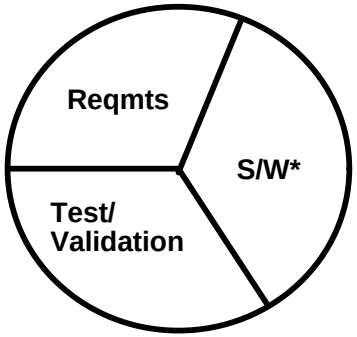
Systems Knowledge

***OMI may require Add'l resources**



S/W Production

- APP S/W Sustaining Support (level of effort %)

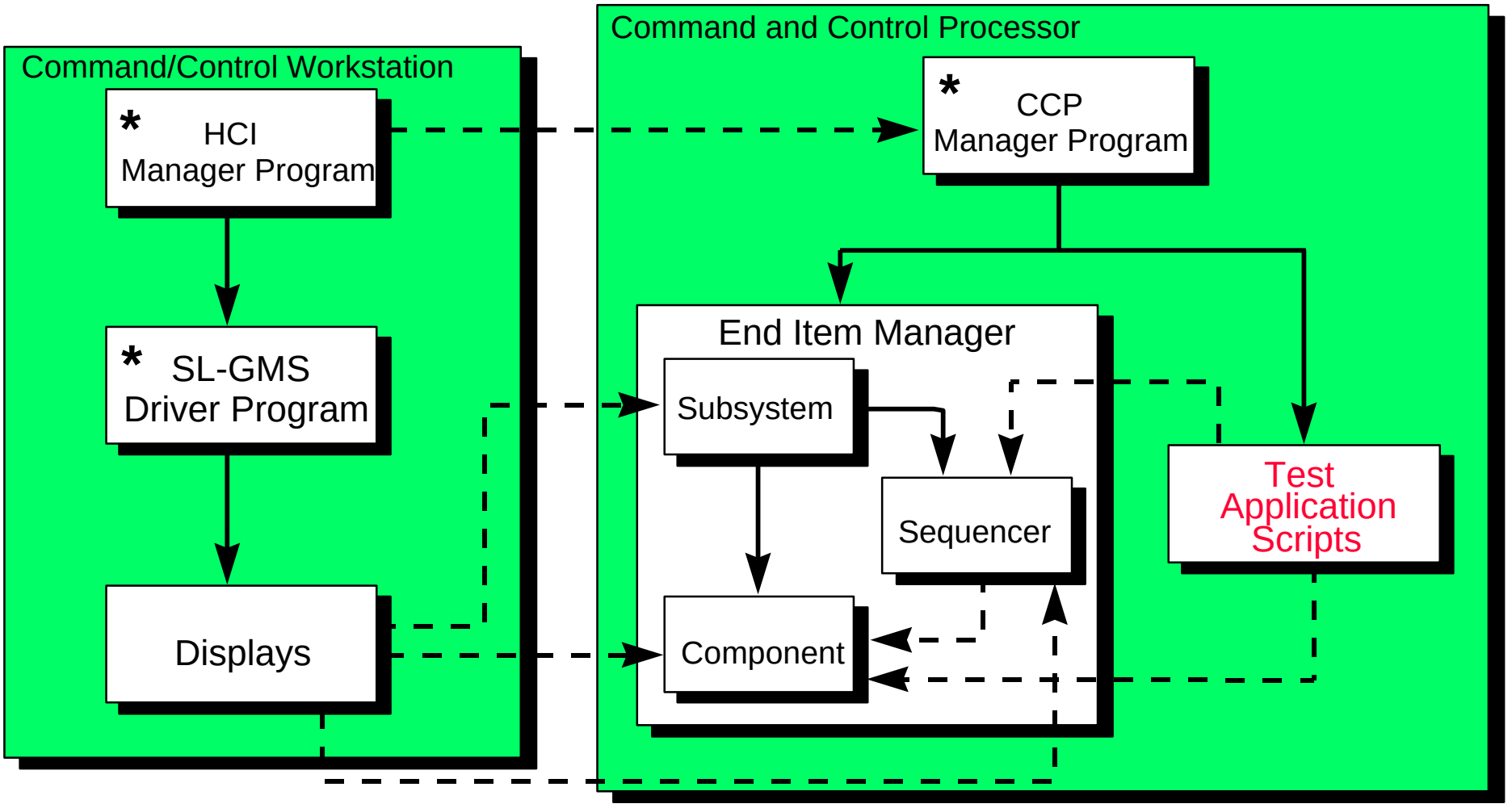


*** S/W implementation of "who" depends on complexity of impact**

Preliminary Estimates

United Space Alliance

APP S/W Set Architecture



Data Health - Data Fusion

- Data Health
  - New method to determine health
  - Data marked as valid or invalid
  - Meaning and Hierarchcy are being worked
  - Methods for Application access and use are being worked

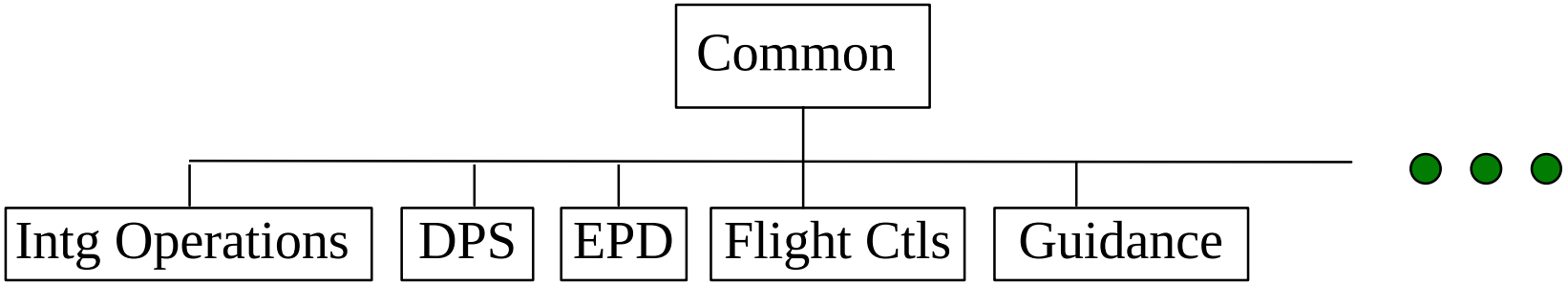
- Data Fusion
  - New class of computed measurements
    - Provide global information (is Main A on?, what is AFT Helium value, etc.)
  - Implementation details are being worked

Deliverables

- Functional Requirements Document
  - Function based
  - New standard format
    - A mentor will guide the IPT on the process
    - This process is the *key* to Application Software mission success
  - Reuse where applicable
  - Must be testable
- Software Specifications
  - Not like CCMS Specs
  - Truly usable by software production organization
  - Format TDB (several options being investigated)
- Application Software
  - Displays
  - HCI Manager Program (HMP) templates (list of programs by subsystem)
  - CCP (Command & Control Processor) Manager Program (CMP) templates
  - End Item Manager
  - Test Application Scripts (TAS functionality is under review)

Documentation

- S/W Documentation Tree
  - New method of document breakdown
    - A “common parts” book has been created to capture re-used items
    - Each subsystem will have its own unique book
  - Provides mechanism to determine scope of change drivers
    - Common parts changes usually are multi-system
    - Changes to a subsystem book typically affects only the system



- Subsystem books
  - Significantly smaller than CCMS requirements
  - IPT’s to define sections in their book(s) (ex: Ground Pwr, Orb main/AC, etc.)

United Space Alliance

Application Components

- Application Development
  - Displays
    - SL-GMS tool of choice
      - Common components available in “shared library”
      - ET monitor effort serves as a pathfinder function
    - Consistent color use across IPTs (common look and feel)
    - May choose to wait as long as possible to develop
  - EIM
    - Control Shell 6.0 has been selected as the tool of choice
      - Finite Sate Machine (FSM) technology
      - C++ based tool
      - Tool is graphical in nature
      - Provides various levels to view a system
      - Various levels of operability (doesn’t require every user to know C ++)
      - Present vision has FSM supporting sequencing
  - TAS
    - Function being evaluated

Validation / User Acceptance

- Simulation
  - Make sure validation requirements are supported
    - Model
    - KATS
    - SAIL
  - May use S/W requirements to identify weak areas
- Validation
  - CLCS project management has identified the levels of testing required:
    1- Development - *Standalone* and *Integrated* (S/W)
    2- Integrated - *Intg Applications Test* (S/W)
    3- Operational - *Validation*, *User Acceptance* (S/W and H/W)
  - Based on requirements
    - Formal and repeatable
    - Use standard format (this process will be refined by the pathfinder IPTs)
    - The "formal" part is the Operational test phase
    - Could be performed by independent person (assuming some system knowledge)

System Knowledge Issues

- Operational procedures (OMI's)
  - Minimize dual maintenance
  - OPF procedures required by June 2000 (start of first CLCS OPF flow)
  - Integrated procedures required by Sept 2000 (first CLCS launch December 2000)
  - Deviations, TPS may be required to demonstrate capability
    - Demonstration requirements will be worked (by project management)
  - Pathfinders will investigate methods to adapt existing procedures
- Training
  - User familiarization training does not require "validated" software
    - May wish to start after displays are available
    - APTeam recognizes concern over "newness"
  - Operational changes
    - Address requirements for Standalone, Cluster and Integrated training
    - Identify "dry run" training requirements
  - Address Certification changes (if any)

Power Up/Down IPT

Robert Pierce (tmp lead)

Requirements

R. Pierce (DPS-tmp focal)
Eric Baker (TPE)
Chuck Rake (EPD)
Nora Lavinka (ECL)
Ron Howard (INS)
Dick Hartung (LPS)
Lisa Schweickart (DPS)
John Apfelbaum (EPD)
Mike Squire (ECL)
Debbie Awtonomow (OPE)
Pete Johnson (INS)
Chuck Klein (LPS)

Procedures
(Validation-OMI)

Same as the requirements personnel

TBD

CORE TEAM

APP Development

Kevin Smith - NASA
Bob Waterman - NASA
Coleman Dugger - NASA

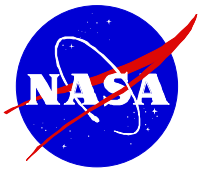
Note: APP development phase starts in January

Alexis Walker (focal)
Kevin Yakubowski
Liz Randall

Simulation

Jim Bunke (focal)

John Allen
Scott Ruch
Lennox Francis



Mentors
Barry Rubel
Larry Leftin

Understudies
Kevin Smith
Bob Waterman

USA
United Space Alliance

IPT Charter

Develop the CLCS Power-Up/Down & Babysit Support APP Software

AND

Perform a pathfinder for the CLCS APP Software Development effort

- The Scope
  - Support the following processes as a minimum
    - Auto power-up and Auto power-down
    - Nominal and Contingency babysit support (not OMI testing)
      - DPS, EPD, ECL (cooling), INS (study if recorders are a candidate)
      - TPE, LPS
  - Provide operability to the following sites
    - OPF, VAB and PAD
- Provide the pathfinder research in
  - Common GSE and Vehicle Components and Operations
  - Evolution of the Display standards (colors, interfaces, etc.)
  - The development process methodology (you are the guinea pigs)

IPT Tasks

1) Refine the Charter
   - The IPT knows the details of the required operability

2) Develop a detailed schedule (TBD's are okay)
   - Requirements
   - Application Software (End Item Mananger, Integrated Sequencing, Displays)
   - Simulation Support
   - Validation, User Acceptance Procedures
   - Operational Procedures (OMI, TPS)
   - Training (How does the IPT plan to teach the new software to users?)

3) Identify resources
   - How many people
     - When are they needed/complete
     - What skills are required
   - Computing resources (if required)

4) Make it Happen

Summary

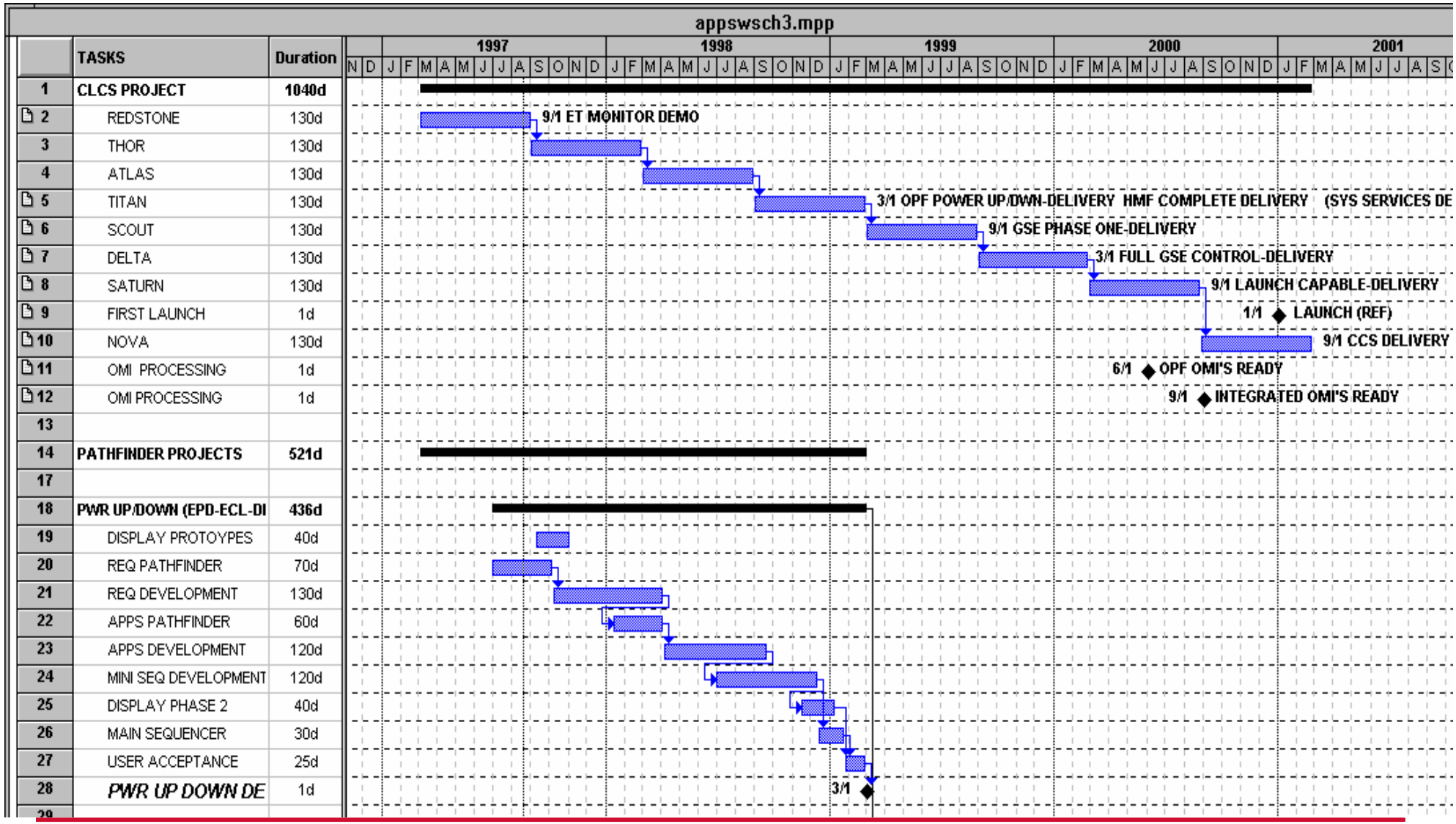
- This ***isn't*** an ASWT (Application Software Working Team)
  - Anyone with available time and ability can work any task
  - Lead is the guide for the overall process (not every process)
  - Lead coordinates through focal points to assure process is on track
  - Focal point works with development group to get the job done
- Tips
  - Remember the big picture while focusing on the task at hand
  - Don't fall into the ruts
  - Don't throw it over the fence
  - Reuse where it makes sense
    - No forcing function to change just for the sake of change
  - The ***team*** will deliver a better product faster
  - Communicate! If there are problems talk to the APTeam (bring issues forward)
  - Don't assume everything is "cast in concrete"
  - The process is iterative and will continue to be refined
- APTeam - IPT tag up 09-10-97
  - Future tag-ups will be held bi-weekly

Backup data

- WWW
  - A CLCS WEB site has been established:http://lpsweb.ksc.nasa.gov/CLCS
  - Application S/W is located at: http://www-clcs.ksc.nasa.gov/mirrors/AppSw
- APTeam e-mail addresses

Robert.Pierce-1@ksc.nasa.gov	861-3427	OSB 4203P
Richard.Ikerd-1@ksc.nasa.gov	861-7541	PCC 2049E
Kenneth.Hale-1@ksc.nasa.gov	861-2261	PCC 3058N

Backup data - IPT targets



	TASKS	Duration
1	CLCS PROJECT	1040d
2	REDSTONE	130d
3	THOR	130d
4	ATLAS	130d
5	TITAN	130d
6	SCOUT	130d
7	DELTA	130d
8	SATURN	130d
9	FIRST LAUNCH	1d
10	NOVA	130d
11	OMI PROCESSING	1d
12	OMI PROCESSING	1d
13		
14	PATHFINDER PROJECTS	521d
17		
18	PWR UP/DOWN (EPD-ECL-DI	436d
19	DISPLAY PROTOYPES	40d
20	REQ PATHFINDER	70d
21	REQ DEVELOPMENT	130d
22	APPS PATHFINDER	60d
23	APPS DEVELOPMENT	120d
24	MINI SEQ DEVELOPMENT	120d
25	DISPLAY PHASE 2	40d
26	MAIN SEQUENCER	30d
27	USER ACCEPTANCE	25d
28	*PWR UP DOWN DE*	1d

United Space Alliance